CS21003 ALGORITHMS-1
(Tutorial 3: Graph)
Date: Sep 19 2020

1 Inconsistent Subset

There are N variables $x_1, x_2, \ldots, x_N$ and M relations of the form $x_i < x_j$ where $i \neq j$. A subset S of relations is called inconsistent if there does not exist any assignment of variables that satisfies all the relations in S. e.g, $\{x_1 < x_2, x_2 < x_1\}$ is inconsistent. You need to find if there is an inconsistent subset of M.

2 Two Coloring

Suppose we have an undirected graph and we want to color all the vertices with two colors red and blue such that no two neighbors have the same color. Design an $O(V + E)$ time algorithm which finds such a coloring if possible or determines that there is no such coloring.

3 Universal Sink

When an adjacency-matrix representation is used, most graph algorithms require time $\Omega(n^2)$, but there are some exceptions. Show that determining whether a directed graph G contains a universal sink (a vertex with in-degree $n-1$ and out-degree 0) can be determined in time $O(n)$, given an adjacency matrix for G.

4 Articulation Points

Let $G = (V, E)$ be an undirected graph. A vertex $v \in V$ is called a cut vertex or an articulation point if the removal of v (and all edges incident upon v) increases the number of connected components in G. Your task is to find all cut vertices in G. What is the running time of this algorithm?

5 Longest Path

You are given an undirected acyclic graph. You need to find a pair of vertices (i, j) such that the length of the path between i and j is maximum among all such pairs. The length of a path is the number of edges on the path. [**Hint:** There is a trivial $O(n.(n + e))$ algorithm to solve this. However, can you give an $O(n + e)$ algorithm?]